

# A SITUATION-RESPONSE MODEL FOR INTELLIGENT PILOT AIDING

Robert Schudy  
Kevin Corker

BBN Laboratories Incorporated  
10 Moulton Street  
Cambridge, Massachusetts 02238

## ABSTRACT

An intelligent pilot aiding system needs models of the pilot information processing to provide the computational basis for successful cooperation between the pilot and the aiding system. By combining artificial intelligence concepts with the human information processing model of Rasmussen, we have developed an abstraction hierarchy of states of knowledge, processing functions, and shortcuts, which is useful for characterizing the information processing both of the pilot and of the aiding system. We are using this approach in the conceptual design of a real-time intelligent aiding system for flight crews of transport aircraft. One promising result from this work has been the tentative identification of a particular class of information processing shortcuts, from situation characterizations to appropriate responses, as the most important reliable pathway for dealing with complex time-critical situations. Situation-response models can be acquired from specialists, such as test pilots and systems engineers, and encoded in a situation-response pilot aiding system. The aiding system can then utilize that specialized expertise to assist flight crews dealing with novel situations, by characterizing the different aspects of the situation, and the appropriate pilot responses, in terms of a finite set of situation types and associated response procedures. There is promise that this approach to aiding will maintain the appropriate level of pilot situational awareness, while maintaining the peak cognitive workloads at levels more characteristic of situation recognition than of problem solving.

## 1. INTRODUCTION

The information available to the pilot of advanced commercial air-transport aircraft is becoming increasingly abstract from the physical parameters of the aircraft that are directly measured and monitored. This is true both for control such as flight controls, and for system monitoring and failure detection mechanisms such as engine monitoring and diagnosis. For example, automatic diagnostic systems have begun to reason about symptoms and situations of fault and failure rather than simply displaying monitored variable values. These trends, are changing the character of the interface between the pilot and the aircraft systems. We have concentrated on the structure of intelligent pilot aiding and pilot interface systems that:

1. Respond to situations such as diagnosis of engine failure;

2. Inform the pilot of these situations (at an adaptable level of detail); and
3. Advise the pilot of actions to be taken in response to the situation.

The architecture of the interface is designed to be quite general in the sense that it will support interactions between a broad range of expert-systems and pilots in a number of types of flight situations; our test cases and examples focus on the interaction and interface management of an engine-fault diagnosis system in commercial air-transports.

One promising result from this work has been the tentative identification of a particular class of shortcuts, from situation characterizations to appropriate responses, as the most important reliable pathway for dealing with complex time-critical situations. Situation-response models can be acquired from specialists, such as test pilots and systems engineers, and encoded in a situation-response pilot aiding system. The aiding system can then utilize that specialized expertise to assist flight crews dealing with novel situations, by characterizing the different aspects of the situation, and the appropriate pilot responses, in terms of a finite set of situation types and response procedures. There is promise that this approach to aiding will maintain the appropriate level of pilot situational awareness, while maintaining the peak cognitive workloads at levels more characteristic of situation recognition than of problem solving. The paper will describe the requirements for intelligent interface management (including the requirements for an explicit model of the pilot information processing functions), and then will outline the implementation architecture designed to meet those requirements.

The example problem being developed in this modeling effort is how the flight-crew and the automatic aiding systems together identify/classify and initiate appropriate response to an engine problem or failure during any portion of a commercial airline flight. Two interactive components of this problem are immediately apparent:

1. The engine diagnosis process (which is being researched at NASA-Langley Research Center); and
2. The selection, communication and execution of appropriate responses for the identified failure in the current context.

We will address the issues of the interface between diagnostic expert systems and the flight-crew, i.e., the selection, communication and initiation of situation information and appropriate responses. The use of the term "appropriate" conveys our concern for the evaluation of the full situation in which engine failure takes place as a necessary condition for response selection and advice. In addition to a full description of the "situation" of engine failure, the fact of cooperation between automatic expert system and human pilots necessitates careful consideration of the processes of human information processing and response selection to assure coordination in cooperation.

## 2. SITUATION-RESPONSE BEHAVIOR

While development of a model for the full repertoire of pilot information processing and flight control behaviors is a task that far exceeds our current state-of-knowledge and technology, we have developed a representation that we feel is appropriate for those behaviors associated with critical time-constrained situations. Our discussions with those responsible for pilot training, and our analysis of National Transportation Safety Board (NTSB) accident reports both lead us to identify a particular human information processing paradigm as predominant and highly preferred for airline pilots when dealing with time-constrained situations. We term this class of information processing *situation-response* behavior. The basic assumptions of the situation-response model are:

- That pilot situation-response information processing involves a situation assessment step in which the current situation is recognized in terms of a finite number of generic situation types; and
- That behavior in response to the situation is driven by procedures previously associated with those situation types.

Before elaborating the specific mechanisms for implementation of this model it is useful to consider the context from which it was derived.

The analysis of systems through description by multi-level abstraction hierarchies is a well established technique (Alexander, 1964, Asimow, 1962). Increasing levels of abstraction provide reduction of physical detail and an increase in functional or goal-oriented specification. It should be noted that the reduction of physical detail as one moves "up" in an abstraction hierarchy is matched by an increase in scope and system-oriented concern for context. More recently, Rasmussen (1983, 1984) has pioneered the description of humans in man/machine systems using the notion of abstraction hierarchies. Specifically, the functions associated with human perception through assessment and response selections and execution have been represented.

Movement through the "perceive/think/act" path (and various shortcuts and heuristics) are presented in Figure 1, which was derived by expressing the abstraction hierarchy in [Rasmussen, 1984] from an artificial intelligence perspective. Figure 1 is based on the description of

processing in terms of an abstraction hierarchy of states of knowledge and processing functions which connect those states of knowledge. The states of knowledge are organized along a horizontal dimension which corresponds to the extent to which the concepts are expressed in terms of the system inputs or in terms of the system response, and along a vertical abstraction dimension. Thus organized, the useful states form a generally triangular shape with the sensors and effectors forming the lower two vertices and the full evaluated set of courses of action the apex. If the representations and processing steps in the sides of this triangle are correct and complete, the the processing sequence from inputs to outputs, following the sides of this triangle, is generally complete and correct. Unfortunately, this path is generally too computationally expensive to be performed in real time, either by natural or artificially intelligent systems. Within the boundaries of the triangle are numerous processing paths which shortcut the detailed processing, by connecting incomplete levels of analysis to partially defined responses. Example shortcuts at different levels of abstraction include reflexes, sensory-motor control, situation-response behavior, and satisficing [Simon, 1969]. The correctness of shortcuts depends on whether the response inferred on the processing shortcut is consistent with the responses which would have been inferred by the computations which are being shortcut. Additional information is provided in the companion paper *A conceptual framework for intelligent real time information processing*, in this volume.

In general, and in the situation-response model, the response for a particular situation is initiated at the lowest level of abstraction which has sufficient scope to select and execute the appropriate response. The situation attributes used to select any one response may span a range of abstraction. For example, the selection of the takeoff abort procedure depends on many higher level attributes such as engine diagnoses, but it also depends critically on the (primitive) air speed attribute. The kind and amount of human information processing required to accomplish a particular behavior is at least as great as that required to generate the highest level abstractions which select that behavior and the most difficult inferences in selecting and executing that behavior. Thus, it makes sense to talk about the level of abstraction of a behavior as a whole. Rasmussen (1983) identifies three general levels of behavioral abstraction:

- "Knowledge-based behavior" in which judgment and decision making and operator models of the system process, contribute to the identification and accomplishment of an operator's goals,
- "Rule-based behavior" by which the characteristics of situation are identified as 'belonging to a set of stored "situations" for which actions and responses are known, but for which procedures need to be tailored to the specific attributes of the situation, and
- "Skill-based behavior" in which limited packets, or sets of behavior are applied to specific stimuli in the environmental situation, with little or no reasoning effort applied to their generation or modification. The approximate ranges of abstraction of these three classes are indicated along the left margin of Figure 1.

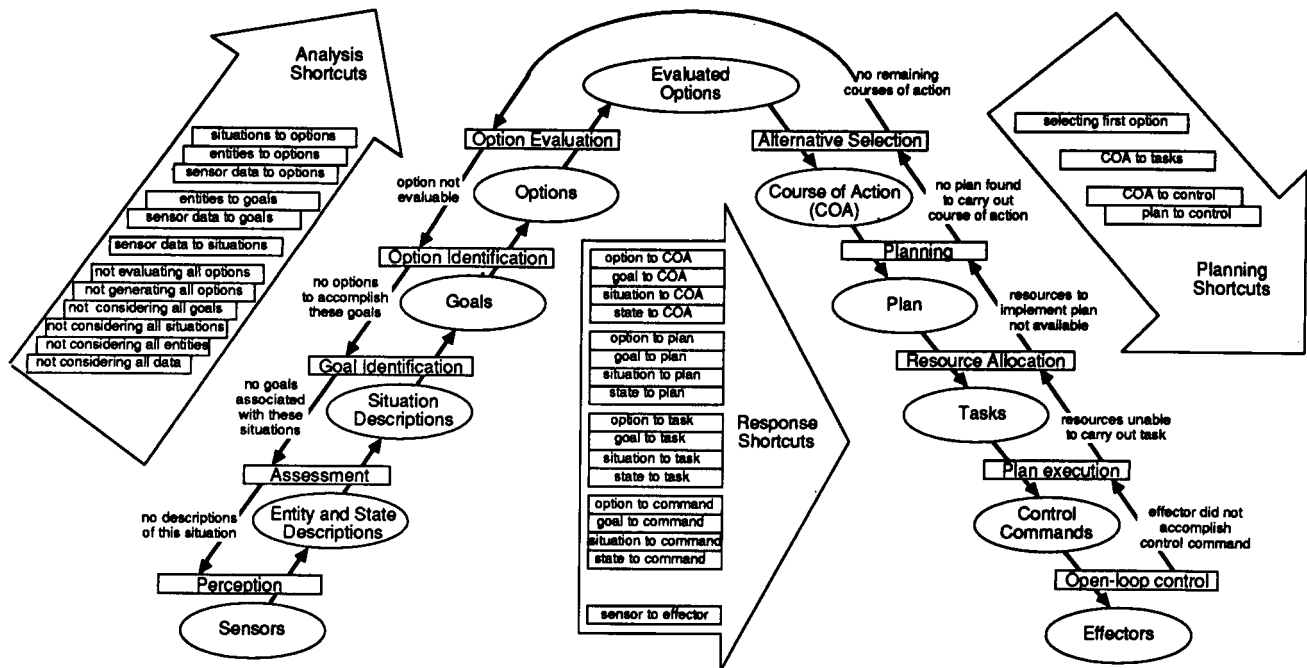


Figure 1. Real Time Processing Abstraction Hierarchy

The most efficient method for identifying an activity to execute is the skill-based strategy, which in the extreme, can be represented as involving no conscious decision-making activity at all, and might even be likened to an automatic reaction to a single stimulus. The correct and efficient enactment of skill-based behavior is expected to take place only after considerable training and/or experience, so that, in some sense, the cost for this efficiency can be thought of as having been borne at a previous time.

The association of behavior with situation attributes directly, without going through the situation assessment process, is a common and useful information processing shortcut. The establishment of such skill-based shortcuts reduces workload and reduces processing delay by uncoupling the situation assessment process from the process of adapting to changing situational parameters. In this model the activation and management of skill-based behavior (e.g., skill-based components of a response procedure) is one of the normal functions of rule-based behavior. The situation assessment function then assumes the role of enabling the execution of the skill-based behavior. Rule-based response selection is represented as taking a greater amount of time to complete, and therefore to tie up the cognitive resources of the pilot for a longer time. The information applied to these types of decisions, like that applied to skill-based decisions, takes the form of a production system. Rule-based decisions differ from skill-based decisions in terms of the number and level of abstraction of the situation attributes which select the behavior response. Rule-based decisions are considered to be more difficult because the enabling conditions are more difficult to compute.

Knowledge-based action selection requires a full analysis of the situation and an assessment of goals before particular

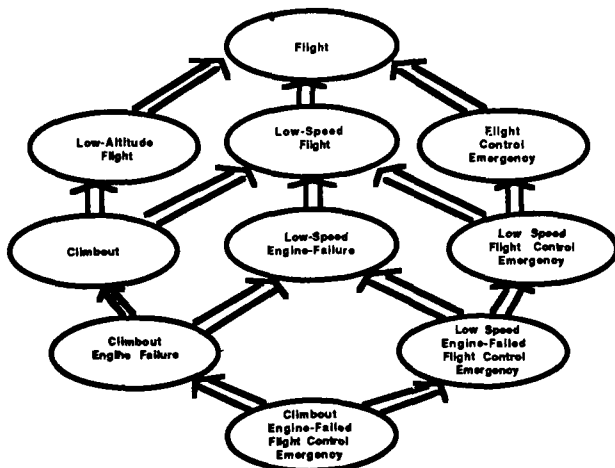
courses of action can be selected and evaluated. Knowledge-based selection typically involves symbolic reasoning processes such as case analysis, projective evaluation, and search. Mental models play a large role in knowledge-based response selection.

Situation-response behavior is the class of rule-based behaviors in which there is a rapid assignment of a response schema to a set of stimuli that have been assembled (through training) into a trigger for the response. Situation-response behaviors are assembled and stored for rapid access and activation without requiring deep or novel reasoning. The links between situation characterization and response initiation are established by processes such as planning, rehearsal, evaluation, trial and error, training and practice. One advantage of situation-response behavior is the efficiency, in terms of time and cognitive resources expended, with which some correct response can be initiated. A second advantage is the ease with which correct situation-response behavioral models can be derived from experiments, experience, and engineering. The disadvantages of situation response behavior lie in the potential for inappropriate situation classification, and in the cost for development and storage of a sufficiently large set of situation types and associated response procedures to adequately deal with a complex and performance-critical task environment.

The focussing of our research on situation-response behavior is motivated by evidence that the need to resort to deep reasoning by aircrews in time-critical flight situations contributes to air transport accidents. Accident analyses suggest that in-flight abstract reasoning may shift attention from flight-critical tasks, and that deep reasoning under stress from potentially incomplete information and incomplete abstract models can produce results which are significantly and sometimes fatally inferior to those derivable from engineering studies, experience, and experiment.

Before engine separation the situation was described in terms of the normal takeoff situation types. At engine separation the engine-loss-climbout situation type also correctly described the situation. The engine-loss-climbout procedures require flight crew attention to airspeed, bearing, climb-rate, thrust-compensation, and crew behaviors designed to compensate for the engine loss and bring the aircraft to a safe altitude and flight path. However, the situation type of engine-loss-climbout did not fully describe the situation. Retraction of the left wing outboard slats placed the flight in a critical stall-regime situation. In low-speed flight any such flight control problem is an emergency of the highest priority, requiring immediate action. The procedures for such low speed

### Example Situation Taxonomy



**Figure 2. Situation Types in the DC-10 Crash**

flight control emergencies are directed toward increasing air speed in order to increase control effectiveness, stall margin, and maneuverability. The appropriate response to this higher priority aspect of the situation would have been to sharply decrease climb angle to gain air speed. To quote the investigatory report:

Each [of these causes: engine loss, slat retraction, warning loss] by itself would not have caused a qualified flight crew to lose control of its aircraft, but together during a critical portion of flight, they created a situation which afforded the flight crew an inadequate opportunity to recognize and prevent the ensuing stall of the aircraft. [NTSB Report #NTSB-AAR-79-17]

The basic problem was that in accepting the initial engine loss situational model, and following the established procedures for situations of that type, the flight crew did not recognize the other more critical aspects of the situation. The challenge for intelligent aiding systems is

being helpful in such emergency situations, when the flight crew doesn't have any resources to spare. Ideally, the aiding system could have prevented the flight crew from accepting the situation as completely described by engine failure on climbout. It is reasonable to assume that a situation assessment system in the aircraft could have quickly detected the anomalous roll by monitoring the flight control and inertial systems. An aiding system could have given behavioral advice, such as maintaining at least V2 air speed, but such unmotivated paradoxical advice might confuse the flight crew. Thus advice such as "Roll Emergency" or "Flight Control Emergency", which implied both the situation and the response, would probably be better. Note that the effectiveness of such communication depends on the flight crew's having models of the emergency situation and associated response procedure, and on the effectiveness of the aiding system in stimulating the appropriate pilot situational awareness and response behavior.

### 3. SITUATION-RESPONSE AIDING

We now describe an approach to aiding the pilot in situation-response behaviors, including the functional requirements for such aiding and an approach to implementing a situation-response aiding system.

### 3.1 Aiding System Function

Figure 3 illustrates the parallelism between pilot situation-response information processing and an intelligent pilot aiding system which is helping the pilot with that processing. The flow at the top of the figure represents the processing steps which an aiding system might go through as it follows the situation-response pathway. The parallel flow at the bottom represents the pilot situation-response processing pathway. The four vertical arrows between these two horizontal flows represent the four main information flows between the pilot and aiding system. The figure shows how aiding systems could assist the pilot in assessing the situation, forming intentions, and executing those intentions. It also illustrates the flow of intentions from the pilot to the aiding system. The following paragraphs describe aiding system functional requirements to support the various phases in the situation-response information processing model.

**Situation type.** Situation types in the aiding system and pilot models should be different for distinguishable situations which require different types of responses. Thus, if two situations which are distinguishable by observable situation attributes have different responses, then those two situation types should be distinct. Conversely, if two situation types are never distinguishable based on their attributes, then the two types should be combined, and behavior associated with the combined type should be appropriate for a situation which could be of either type. Similarly, if two situations may or may not be indistinguishable, based on situation attributes, and those situations have different behaviors, then it is usually appropriate to generate an additional more general situation type which models the uncertainty by spanning the set of situation types which cannot be distinguished, based on situation attributes. The behavior associated with this more general situation type should be appropriate for the state of knowledge of the situation. When more specific situation attribute knowledge is available, then one of the more specific types should be used.

**Pilot mental model.** Effective high-level communication between intelligent aiding systems and the pilot requires compatibility between the conceptual model implemented by the intelligent aiding system at the pilot interface and the conceptual model held by the pilot. That is, the system image must be compatible with the pilot mental models. Because effective communication about complex topics depends on coherent relationships between topics at multiple levels of abstraction, severe requirements are imposed on the compatibilities between the images presented by intelligent pilot aiding systems and the pilot's own mental models. Indeed, it may be necessary, or at least desirable, to base the system image on an explicit representation of prototype pilot mental models.

**Situation Attributes.** The human interface should provide sufficient information on the situation attributes, at the appropriate level of detail, and in the appropriate structure, so that the pilot can correctly classify the situation in terms of his mental situation models. Different values of situation attributes which are significant in terms of situation assessment and response should be clearly distinguishable to the pilot who is to base his situation model and response on those attributes. The systems should provide the situation attributes at a level of detail which matches the human information processing input requirements. For situation-response processing this level of detail is the level necessary to unambiguously identify the situation type and to refine the attributes of that type at the level of detail needed to support the response. It is clearly critically important that situation descriptions be refined to the level of detail required to support the correct pilot response. It can also be very important that superfluous detail not overload and distract the pilot. The system should therefore provide different levels of detail and different foci of detail to support changing human information requirements.

**Assessment aiding.** The human computational burden of situation assessment and situation classification can be reduced by automating some of the assessment functions. The processing to support this assessment involves the fusion of different situation attributes into more abstract attributes which support a simpler form of the situation assessment processing. For example, a pilot may be presented with N1, N2, temperature, pressure and other engine information. He may also be presented with airspeed, altitude, throttle and other information. What the pilot needs to know is how the engine is performing now and how it will perform in the rest of the flight. An engine diagnosis system should combine this wealth of data from the sensor systems, to produce a model of the engine status which can be more easily matched to situation types which serve as the basis for pilot action. Intelligent pilot interface and aiding systems should also support the pilot in giving priority to the assessment of critical aspects of the situation.

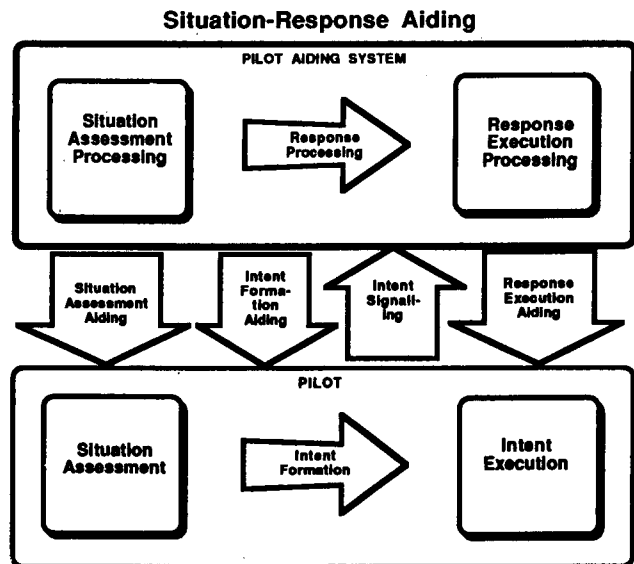


Figure 3. Situation-Response Aiding

**Response Resolution.** When different aspects of the situation require different procedures, then aiding systems can support the pilot's resolution of this behavioral dissonance, by constructing or identifying appropriate response procedures. These procedures can generally be derived from procedures for the different aspects of the situation. For example, an aiding system may be able to determine that a particular reduced throttle schedule on a damaged engine would provide the most favorable compromise between the conflicting behaviors of obtaining maximum thrust and protecting against total engine failure.

**Knowledge-Based Reasoning.** Experience with air crashes and other accidents indicates that pilot deep reasoning in dangerous situations with real-time response requirements has some undesirable properties. For example, successful knowledge-based reasoning normally requires case analysis, comparison with examples, mental simulation, and other cognitive activities which consume most or all of the human cognitive capabilities for significant time intervals. Further, knowledge-based reasoning under stress in real-time situations often leads to erroneous shortcuts in analysis. Thus automation of the knowledge-based reasoning to support situation-response behavior of pilots is highly desirable. It is also highly desirable that the aiding system not impose the requirement for knowledge-based reasoning, either through not fully supporting operator situation-response behavior, or through requiring knowledge-based behavior to use the aiding system.

**Feedback.** Humans or automatic controllers often require some information on the state of whatever it is they are effecting in order to monitor the progress of the response, and to provide the feedback necessary to implement any control laws in the response. The nature of the feedback can be determined using models of human performance and through experiment. Considerable benefit can be gained by adapting the feedback to the situation, response, and control laws. Our pilot information processing model suggests, through its hierarchical structure, that both control and feedback be available at multiple levels of the abstraction hierarchy. The particular form of the feedback or control should be geared to the level at which the pilot is interacting with the system.

**Goal Monitoring.** Responses to situations, either by automated systems or by the pilot, usually have as their focus attainment of some goal which terminates the tasks (or subtasks) required for that goal's achievement. If an intelligent aiding system is aware of tasks focused on a particular goal, and can determine when the goal state has been reached, then it may be of considerable value to the flight crew to announce the attainment of those goal states. Alternatively, a task may be one in which the goal state is to be maintained until some expected event occurs. Intelligent aiding systems can provide valuable assistance by inferring the values and ranges of those goal states, signalling their initial attainments, monitoring for their maintenance, and responding appropriately when the termination event occurs.

**Context Change Monitoring.** A task may become inappropriate because the context in which the task was being performed has changed sufficiently to make the task impossible to perform or to make the goals no longer of interest. The pilot information processing model provides two general pathways through which this can be discovered. One pathway follows from situation assessment. The second pathway follows from the monitoring of the ongoing task execution, when task execution requirements are no longer met, task performance expectations are not met, or other unexpected conditions are discovered. Intelligent aiding systems can assist in these cases by assessing the situation and monitoring for conflicts, and by monitoring task execution and appropriately alerting the flight crew of tasks which must be modified, abandoned or replaced.

### 3.2 Aiding System Implementation Approach

A computer implementation of the situation-response information processing model, using machine intelligence techniques, is illustrated in Figure 4.

The upper three boxes hold examples of the *a priori* knowledge structures for situation attributes, situation types, and response procedures. The lower three boxes hold examples of the runtime instances of those situation attributes, situations, and procedures. Arcs in the figure illustrate the explicit relations between the representations, and the large arrows illustrate the runtime processing steps. In our baseline implementation approach all six of these knowledge structures are represented by frames. The following paragraphs describe the representations and processing steps in Figure 4.

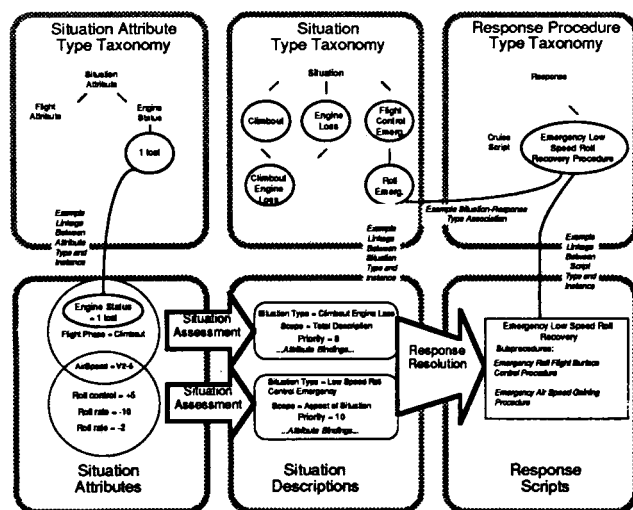


Figure 4. Overview of Situation-Response Implementation

**Situation Types.** Situation types are attempts to represent the pilot's mental models of generic classes of situations. For example, preflight, cruise, and landing roll are different types of situations in a typical flight. A situation type is represented in our computer model as a frame in a frame-based knowledge representation system. (In a frame-based representation, the frames consist of sets of ordered pairs of slot descriptions and slot fillers. The slot descriptions specify the relationship of the slot filler to whatever is being described by the frame; the slot descriptions are therefore often termed "relations" or "roles.") In situation type frames the slot is filled with a description of the class of things which could fill this slot in an instance of this type of situation. Situation types are sets of descriptions of relevant attributes of the generic situation and the generic relationships between those attributes. In the situation-response model each situation type may have associated with it a description of the behavior to be performed in that situation. For example, an engine-failed-climbout situation type might be described by the frame:

```
engine-failed-climbout
  SuperC      climbout-situation
  SuperC      single-engine-failure-situation
  response    engine-failed-climbout-procedure
```

The SuperC relations indicate that this situation is a specialization of both the climbout situation and the situation in which a single engine has failed. If the engine-failed-climbout situation were described as a specialization of climbout-situation and of single-engine-failure-situation all of the slots of those situations (and the SuperCs of those situations) would be *inherited* by the engine-failed-climbout situation.

**Situation Response Procedures.** Associated with each situation type is one description of the procedure to be performed in situations of that type. These procedures use the actual values of situation attributes in much the way that a computer software procedure uses the formal parameters. For example, the final approach procedure may key specific actions to specific values of the altitude attribute of the final approach situation.

**Situation Assessment.** Situation assessment can be modelled computationally by a matching or classification process in which the perceived situational attributes form a pattern, and the goal is finding all the situation types which can fit that pattern. Note that situation assessment does not attempt to resolve the ambiguities and inconsistencies due to the lack of information. Indeed, it cannot do so reliably. Rather, situation assessment provides a description of the *possible* interpretations of the current situation, together with the assumptions underlying those interpretations.

**Perceived Situation Attributes.** Perceived situation attributes are the attributes of the situation which are computed in real time from sensor data and models of the things perceived. This processing can be hierarchically structured, as sensor information is combined into abstractions with successively larger scope. For example, perceptual processing may include diagnosis of an incipient engine failure.

*Situation Descriptions.* The situation assessment process produces a situation description from an appropriate situation type, by replacing the attribute descriptions of the situation type with their refined values, to produce an instance of the situation type tailored to the actual current situation. For example, a description of a single engine failure situation might be derived from the single-engine-failure situation type, in part, by replacing the failed-engine ID attribute of the type by the ID (e.g., left, right) of the failed engine.

Ideally all situations could be described in terms of situation types which accounted in detail for all of the attributes of the situation. Such a well-fitting situation type would result in an equally appropriate situation description, and a very well focused response procedure. In some domains the number of different kinds of situations, and the number of combinations of different situation attributes, may be sufficiently small to permit the tailoring of situation types to each of the combinations of situation attributes. However, in complex domains, the number of different combinations of situation attributes precludes unique association of a situation type with each possible combination of attributes. There are at least four ways to obtain reasonable situation descriptions without having an unmanageable number of situation types: by refining attributes using perceived values, by using more general situation types, by describing the situation in terms of its different aspects, and by describing situations with subsumption hierarchies of descriptions of different levels of detail.

#### 4. Conclusions

When the information processing pathways of a pilot or intelligent aiding system is laid out in an abstraction hierarchy stretching from inputs to actions, a particular subset of those pathways is found to describe the most important and desirable for pilots pilots others engaged in critical time-constrained system operation tasks. A model of these situation-response behaviors forms a sound basis for pilot training and for systems which aid the pilot in correctly assessing and responding to situations. Major research tasks remaining include verifying the scope of the model relative to the full range of pilot aiding requirements, implementing a situation-response aiding system, and testing with pilots in realistic real-time situations.

#### 5. REFERENCES

Alexander, C., Notes on the Synthesis of Form. Harvard Press, Cambridge, MA, 1964.

Asimow, M., Introduction to Design Prentice-Hall, Englewood Cliffs, NJ, 1962.

Rasmussen, J., Skills, Rules, Knowledge; Signals, Signs and Symbols; and other Distinctions in Human Performance Models. IEEE Transactions of Systems, Man, and Cybernetics SMC-13 (3), 1983.

Rasmussen, J., Strategies for State Identification and Diagnosis in Supervisory Control Tasks, and Design for Computer-based Support Systems. In (Rouse) Advances in Man-Machine Systems Research, Vol. 1 pp. 139-193, 1984.

Simon, H., The Sciences of the Artificial. M.I.T Press, 1969.